

**SOURCES DE PROGRAMMES
D'ANALYSE DE DONNÉES EN LANGAGE PASCAL:
(III) : ÉLABORATION DE TABLEAUX DIVERS
(IIID) : PRÉSENTATION GRAPHIQUE DES
LIGNES D'UN TABLEAU**

[SOURCES PASCAL (IIID)]

J.-P. & F. BENZÉCRI

0 Introduction : fonction du programme Grftab

On rencontre communément des tableaux de données où l'ensemble J des colonnes n'est autre qu'un ensemble T de temps: dates ou instants régulièrement espacés; tandis que chaque ligne enregistre la variation temporelle d'un phénomène, à l'échelle fixée par T. Il peut être utile de préparer ou d'illustrer l'analyse multidimensionnelle par des courbes figurant les données d'une ligne; soit à l'état brut; soit peu élaborées, e.g., sous forme de profil; en adoptant, éventuellement, une même échelle pour toute les courbes.

Il va sans dire que le tracé de courbes est une fonction commune à tous les tableurs; mais la notion de profil individuel, considéré par rapport au profil marginal, notion fondamentale, pour nous, tant en analyse de correspondance qu'en CAH, n'est pas directement accessible dans les logiciels en usage. Aussi publions-nous ci-après le programme 'Grftab', non comme un produit achevé, mais comme un exemple des fonctions dont nous souhaitons voir doté le tableur de l'avenir.

L'exposé ci-après suit l'ordre du listage; en détaillant seulement les fonctions particulières et les choix de format qu'il nous a paru utile d'introduire. Nous considérons successivement: la désignation du tableau de base, éventuellement restreint à un de ses sous-tableau (§1); la spécification du format des graphiques (§2); et le tracé proprement dit (§3).

1 Choix et restriction du tableau de base

1.1 Structure du programme

Après des déclarations destinées à loger le tableau considéré et sa marge horizontale, avec diverses bornes et formats, vient le programme, compris tout

```

program grftab;
uses memtypes, quickdraw, osintf, toolintf, sane, uver5, ulire5;
const jsup=150;
var i, ii, ic, jr, ja, jb, jc, j, ni, n1, n2, n3, carti, cartj, ia, ib, jja, jjb,
    Uch, Ucv, Qch, Qcv, lla, hha, col, pas, frac, frc, carc: integer;
max, min, mxg, mng, tot, co2: extended;
sgli, sg2j: pzgi; k1ji: kji; mrj: array[1..jsup] of real;
nomba, nomf, titre, sig: string;
repc, rpf, rpb, rpc, rpt, rpp, rpq, rps, rpi, rpj, rpa, rpv, rpd: char;
tlk: tk; sli, s2j: ptr; rec0, recl: rect;
begin; benzecri;
writeln('ce programme présente les courbes des lignes'); repc:='O';
while not (repc='N') do begin carti:=1;
    litab(s2j, @tlk, @nomba, @titre, repc, carti, cartj, sli);
    if ((repc='O') and (cartj<3)) then begin repc:='N';
        for j:=0 to cartj do dispose(tlk[j]); dispose(sli); dispose(s2j); end;
    if (repc='O') then begin

```

entier dans une boucle: while not(repc='N') do...; laquelle débute par la procédure 'litab' de choix et d'entrée du tableau de base (cf. IA§2). On notera que 'litab' est appelée ici avec: carti=1, ce qui revient à réserver la place d'une ligne de plus: la marge, dont on suivra le calcul au §2.

Si l'utilisateur s'est fixé sur un tableau accessible, et ayant au moins trois colonnes, le traitement se fait dans la boucle: if (repc='O') then begin... Après un dialogue offrant de nombreux choix de formats, vient l'affichage des courbes; et, quand celui-ci est terminé, l'utilisateur peut, par la réponse: repc='O', demander à rentrer dans la boucle générale pour considérer un nouveau tableau.

```

    ia:=1; ib:=carti; rpi:='O'; ja:=1; jb:=cartj; rpj:='O'; frac:=1;
    for j:=1 to cartj do k1ji[j]:=pti(tlk[j]);
    sgli:=pzgi(sli); sg2j:=pzgi(s2j);
    setrect(rec0, 0, 0, 550, 350);
    writeln(titre); writeln('carti = ', carti:4, ' ; cartj = ', cartj:4);
    writeln('NB l'ensemble J des temps peut être restreint à un sous-ensemble');
    writeln('d'une part, en reserrant l'intervalle par des bornes');
    writeln('d'autre part, en ne prenant qu'une fraction des éléments');
    if (jsup<cartj) then begin rpj:='O';
        writeln('Dans le cas présent, cardJ = ', cartj:4);
        writeln('il doit être réduit à, au plus ', jsup:4, ' '); readln(rpa); end;
    if (cartj<=jsup) then begin
        write('faut-il restreindre l'ensemble J, O ou N ');
        readln(rpj); end;

```

1.2 Restriction de l'ensemble J des temps (ou abscisses des courbes)

Dans la version publiée par nous, la procédure 'litab' fixe à: jmax=300, le nombre maximum de colonnes des tableaux traités. Pour 'Grftab', le nombre maximum des points d'un graphique est fixé à: jsup=150. Si cartj dépasse 150, l'utilisateur doit restreindre l'ensemble des temps à un sous-ensemble de J.

À cette fin, sont proposés deux procédés, qu'on peut employer simultanément: d'une part, restreindre à [ja..jb] l'amplitude des temps; d'autre part, ne figurer qu'un point sur 2; ou un sur 3, etc. Éventuellement, on restreindra J même si: cartj ≤ jsup=150.

```

while not (rpj='N') do begin
  eraserect(rec0);moveto(2,20);rpd:='N';
  writeln('ci-dessous sont rappelés les num et sigles des colonnes');
  for j:=1 to cartj do begin write(j:4,sigler(sg2j^[j]):5,');
    if (j mod 8=0) then writeln;
    if (j mod 200=0) then begin
      write('pour afficher la suite entrer * ');readln(rpb);end;end;
  if not (cartj mod 8=0) then writeln;
  write('le numéro de la 1-ère colonne de l''intervalle retenu est ');readln(ja);
  write('le numéro de la dernière colonne de cet intervalle est ');readln(jb);
  if (ja<1) then ja:=1;if (cartj-2<ja) then ja:=cartj-2;
  if (jb<ja+2) then jb:=ja+2;
  if (cartj<jb) then jb:=cartj;
  writeln('bornes: ja = ',ja:4,' ; jb = ',jb:4);
  if (jsup<jb+1-ja) then begin rpd:='O';
    frac:=(jb+1-ja) div jsup;
    if not ((jb+1-ja) mod jsup =0) then frac:=frac+1;
    writeln('cet intervalle devra être réduit à une fraction de ces éléments');
    writeln('soit: f=1/frac, avec frac≥ ',frac:2);end;
  if (rpd='N') then begin frac:=1;
    write('faut-il réduire l''intervalle à une fraction de ses éléments, O ou N ');
    readln(rpd);if not (rpd='N') then rpd:='O';end;
  if (rpd='O') then begin
    write('la fraction, f=1/frac, choisie est définie par frac = ');
    readln(frc);if (frac<frc) then frac:=frc;
    if ((jb+1-ja) div frac <2) then frac:=(jb+1-ja) div 2;
    writeln('frac = ',frac:3);end;

  write('ces choix sont-ils confirmés O ou N ');readln(rpa);
  if (rpa='O') then rpj:='N';end;
  jja:=ja;jjb:=jb;
  if (1<frac) then begin carc:=0;
    for j:=ja to jb do if ((j-ja) mod frac =0) then begin
      carc:=carc+1;klji[carc]:=klji[j];jjb:=j;end;
    ja:=1;jb:=carc;end;

```

La boucle: while not(rpj='N') do begin..., a pour fonction de restreindre l'intervalle J des temps; que le choix: rpj='O' soit imposé par le programme (si: cartj>jsup=150), ou demandé par l'utilisateur. Le taux d'échantillonnage est noté: frac.

On retiendra que les bornes définissant l'amplitude des temps, d'abord notées: ja, jb; sont recopiées dans: jja, jjb. Car si: frac≠1, (i.e. si, de frac instants consécutifs, un seul est retenu), il est commode de renuméroter consécutivement, de 1 à carc, les pointeurs afférents aux colonnes conservées; et alors, en posant: ja=1, jb=carc, on fait en sorte que l'affichage des courbes, aille, dans tous les cas, prendre de ja à jb, les coordonnées des points retenus.

1.3 Restriction de l'ensemble I des lignes à représenter

Restreindre l'ensemble I, c'est-à-dire l'ensemble des lignes dont l'utilisateur peut demander un graphique, ne s'impose pas au même titre que de restreindre l'intervalle J des temps. Mais si (cf. *infra*) les courbes représentent des profils, il vaut la peine de choisir la partie de I sur laquelle est calculée, par sommation, la marge horizontale. Et lorsque tous les graphiques

```

write('faut-il restreindre I à un sous-intervalle O ou N ');
readln(rpi);
while not (rpi='N') do begin
  eraserect(rec0);moveto(2,20);
  writeln('ci-dessous sont rappelés les num et sigles des lignes');
  for i:=1 to carti do begin write(i:4,sigler(sgli^[i]):5,' ');
    if (i mod 8=0) then writeln;
    if (i mod 200=0) then begin
      write('pour afficher la suite entrer * ');readln(rp);end;end;
  if not (carti mod 8=0) then writeln;
  write('le numéro de la première ligne retenue est ');readln(ia);
  write('le numéro de la dernière ligne retenue est ');readln(ib);
  if (ia<1) then ia:=1;if (carti<ia) then ia:=carti;
  if (ib<ia) then ib:=ia;if (carti<ib) then ib:=carti;
  writeln('bornes: ia = ',ia:4,' ; ib = ',ib:4);
  write('ces bornes sont-elles confirmées O ou N ');readln(rpa);
  if (rpa='O') then rpi:='N';end;

```

sont tracés à la même échelle, il peut être nécessaire d'en éliminer certains, pour que les autres restent lisibles.

2 Spécification du format des graphiques

2.1 Les choix de l'utilisateur

```

write('faut-il faire des graphiques grands(G) ou petits(P) ');
readln(rpt);lla:=470;hha:=244;
if (rpt='P') then begin lla:=156;hha:=120 end;
write('faut-il représenter lignes brutes(B) ou les profils/marge(P) ');
readln(rpp);if not (rpp='P') then rpp:='B';
if (rpp='P') then begin
  writeln('faut-il, pour le calcul de la marge, pondérer les lignes');
  write('afin qu'elles aient toutes un pour total O ou N ');readln(rp);
  if not (rp='O') then rp:='N';end;
write('faut-il tracer tous les graphiques à la même échelle O ou N ');
readln(rps);if not (rps='N') then rps:='O';
write('faut-il tracer des barres verticales(V) dans les graphiques ou non(N) ');
readln(rp);

```

Ont paru dans *CAD* des articles illustrés de planches portant de nombreux graphiques de petite taille, saisis successivement, et rassemblés pour offrir une vue d'ensemble. Mais parfois, on demandera un grand graphique afin de ne rien perdre des détails; surtout si des courbes diverses doivent être rendues à la même échelle; ou que l'utilisateur s'applique à découvrir les données, sans prétendre saisir à l'écran des graphiques destinés à la publication.

On a déjà fait allusion à la présentation des lignes par leurs profils (plus précisément, le profil de chaque ligne est rapporté à celui de la marge). Au §1.3, à propos de la restriction de I, on a évoqué le cas d'un ensemble I de lignes hétéroclites; dont certaines nuisent à l'examen des autres. Sans éliminer de lignes, on peut restreindre l'effet de lignes aberrantes, de très fort poids, en réduisant les lignes du tableau de base à avoir toutes le même total.

Le tracé de barres verticales (régulièrement espacées sur J), aide à faire voir nettement l'échelle des temps; mais, parfois, trop de traits surchargent le graphique.

```

i:=ia-1;
while ((rpp='P') and (i<ib)) do begin i:=i+1;tot:=0;
  for j:=ja to jb do begin
    if (klji[j]^i<0) then rpp:='b';tot:=tot+klji[j]^i;end;
    if (tot=0) then rpp:='n';end;
  if (rpp='n') then begin rpp:='B';
    writeln('ERREUR, la ligne',sigler(sgli^i):5,' a un total nul');
    write('on ne tracera donc pas de profil ');readln(rpf);end;
  if (rpp='b') then begin rpp:='B';
    writeln('ERREUR, la ligne',sigler(sgli^i):5,' comporte un nombre négatif');
    write('on ne tracera donc pas de profil ');readln(rpf);end;
  j:=ja-1;
  while ((rpp='P') and (j<jb)) do begin j:=j+1;mrj[j]:=0;
    for i:=ia to ib do mrj[j]:=mrj[j]+klji[j]^i;
    if (mrj[j]=0) then rpp:='b';end;
  if (rpp='b') then begin rpp:='B';
    writeln('ERREUR, la colonne',sigler(sq2j^j):5,' est nulle');
    write('on ne tracera donc pas de profil ');readln(rpf);end;

```

2.2 Vérifications pour le calcul des profils

On ne calcule de profils que si aucune des lignes retenues (cf. §1.3) ne comporte de nombre négatif ou n'a un total nul. De plus, en calculant la ligne de marge, on vérifie que celle-ci ne comporte pas de zéro.

2.3 Élaboration du tableau et préparation des tracés

```

if (rpp='P') then begin
  for i:=ia to ib do begin tot:=0;
    for j:=ja to jb do tot:=tot+klji[j]^i;tot:=1/tot;
    for j:=ja to jb do klji[j]^i:=tot*klji[j]^i end;
  if (rpp='O') then for j:=ja to jb do begin mrj[j]:=0;
    for i:=ia to ib do mrj[j]:=mrj[j]+klji[j]^i;end;
  sgli^[carti+1]:=zigler('Marg');tot:=0;
  for j:=ja to jb do tot:=tot+mrj[j];tot:=1/tot;
  for j:=ja to jb do begin
    mrj[j]:=tot*mrj[j];klji[j]^i:=mrj[j];end;

  for j:=ja to jb do begin tot:=1/mrj[j];
    for i:=ia to ib do klji[j]^i:=tot*klji[j]^i;end;end;

```

S'il faut représenter des profils, chaque ligne retenue est d'abord divisée par son total. Si l'utilisateur a choisi de donner même poids total à toutes les lignes (rpp='O'; cf. §2.1), la ligne de marge est recalculée comme somme des lignes ainsi modifiées. Finalement, est adjointe au tableau une ligne dont le sigle est: Marg, donnant le profil de la marge retenue; et à ce profil sont rapportés ceux des lignes retenues (numérotées de: ia à ib). Les quotients: (profil/marge), constituent le contenu définitif du tableau: klji.

Du rectangle où doit s'inscrire le tracé de chaque courbe, la largeur: lla, est partagée en autant de bandes de largeur: col (en pixels), qu'il y a d'intervalles entre les instants représentés: col:=trunc(lla/(jb-ja)); éventuellement, lla est corrigée. La hauteur: hha, vaut, en principe, les 3/4 de la largeur; mais elle limitée à 244. Il est clair que ces formats, adaptés aux plus petits écrans des Macintosh, n'ont rien d'essentiel.

```

col:=trunc(11a/(jb-ja));if (col<3) then col:=3;
11a:=col*(jb-ja);hha:=trunc(3*11a/4);if (244<hha) then hha:=244;
pas:=1;if (col<8) then pas:=2;if (col<4) then pas:=3;
mxg:=0;mng:=0;
if (rps='O') then for j:=ja to jb do for i:=ia to ib do begin
  if (mxg<klji[j]^i) then mxg:=klji[j]^i;
  if (klji[j]^i<mng) then mng:=klji[j]^i;end;
rpc='O';
setrect (rec1,2,20,3+11a,21+hha);

```

Enfin, en vue du tracé de courbes toutes à la même échelle, ont déterminé le maximum général: mxg, et le minimum: mng, de l'ensemble des valeurs se rencontrant dans les lignes retenues; i.e., des ordonnées des points des courbes.

Choix et calculs préliminaires étant ainsi achevés, on entre dans la boucle de tracé des courbes.

3 Le tracé des courbes

```

while not (rpc='N') do begin
  eraserect (rec0);moveto (2,20);
  writeln ('ci-dessous sont rappelés les num et sigles des lignes');
  for i:=ia to ib do begin write (i:4,sigler (sgli^[i]):5,',');
    if ((1+i-ia) mod 8=0) then writeln;
    if ((1+i-ia) mod 200=0) then begin
      write ('pour afficher la suite entrer * ');readln (rpb);end;end;

  if not ((1+ib-ia) mod 8=0) then writeln;
  if (rpp='P') then writeln ('NB pour afficher la ligne de marge entrer 9999');
  write ('le numéro de la ligne à afficher est ');readln (i);
  if (i<ia) then i:=ia;
  if ((ib<i) and not (rpp='P')) then i:=ib;
  if ((ib<i) and (rpp='P')) then i:=carti+1;

```

3.1 Choix de la courbe à tracer

Sur la suite des sigles des lignes retenues, l'utilisateur choisit d'abord la ligne à afficher; à moins qu'il ne s'agisse de la ligne de marge. Si le numéro demandé sort des limites permises, il est corrigé.

3.2 Calcul de l'échelle verticale et tracé du cadre

La courbe s'inscrit dans le cadre rectangulaire: rec1, défini au §2.3.

```

eraserect (rec0);penmode (pator);framerec (rec1);
min:=0;max:=0;
if ((rps='O') and not (i=carti+1)) then begin min:=mng;max:=mxg;end else
  for j:=ja to jb do begin
    if (max<klji[j]^i) then max:=klji[j]^i;
    if (klji[j]^i<min) then min:=klji[j]^i;end;
if (min=max) then co2:=0 else co2:=hha/(max-min);
if not (min*max=0) then begin
  Ucv:=20+round (co2*max);moveto (2,Ucv);lineto (11a+2,Ucv);end;
if ((rpp='P') and not (i=carti+1)) then begin
  Ucv:=20-round (co2*(1-max));
  moveto (2,Ucv);lineto (11a+2,Ucv);end;
if not (rpv='N') then for j:=ja+1 to jb-1 do
  if ((j-ja) mod pas=0) then begin
    Uch:=2+(col*(j-ja));moveto (Uch,20);lineto (Uch,hha+20);end;

```

N.B.: Pour interpréter les ordres de tracé, le lecteur, doit se souvenir que, selon les conventions du Macintosh, la coordonnée verticale de l'écran est comptée en croissant du haut vers le bas.

Le coefficient: $co2$, de l'échelle verticale, est calculé, selon les cas, en fonction de l'amplitude générale de variation de l'ordonnée: [mng..mxg], ou de l'amplitude: [min..max], propre à la courbe choisie.

Une ligne horizontale est tracée à la distance: max, du bord supérieur: si: $min < 0 < max$, cette ligne définit le niveau zéro; sinon, elle se superpose au bord inférieur du cadre. Dans le cas où la courbe représente un quotient: (profil/marge), on trace, de même, une ligne horizontale au niveau 1. Si l'utilisateur l'a demandé, sont tracées, du bord supérieur au bord inférieur du cadre, des lignes verticales régulièrement espacées.

3.3 Tracé de la courbe comme une ligne brisée

```
for j:=ja to jb do begin
  Qch:=2+(col*(j-ja));Qcv:=20-round(co2*(k1ji[j]^i-max));
  if (j=ja) then begin moveto(1,Qcv);lineto(2,Qcv);end;
  if (ja<j) then lineto(Qch,Qcv);
  if (j=jb) then lineto(Qch+1,Qcv);
  Uch:=Qch;Ucv:=Qcv;moveto(Uch,Ucv);end;
```

On notera seulement que le début et la fin de la courbe sont marqués par un pixel, en dehors du bord du cadre.

```
penmode(patXor);moveto(4,10);
if not(rpv='N') then begin
  frc:=pas*frac;drawstring('pas = ');drwnum(frc);drawstring(' ');end;
drawstring(' haut= ');drwfl(max);
if (rpp='P') then drawstring(' ; bas=0 ') else
  begin drawstring(' ; bas= ');drwfl(min);end;
if ((rpp='P') and not(i=carti+1)) then drawstring(' ; profil/marge ');
if (i=carti+1) then drawstring(' ; ligne de marge ');
if (rpp='P') then begin
  drawstring('de lig n°');drwnum(ia);
  drawstring(concat(':',sigler(sgli^[ia])));
  drawstring(' à lig n°');drwnum(ib);
  drawstring(concat(':',sigler(sgli^[ib])));end;
moveto(4,hha+30);
drawstring(concat(nomba,' ; de la col n°'));
drwnum(jja);drawstring(concat(':',sigler(sg2j^[jja])));
drawstring(', à la col n°');
drwnum(jjb);drawstring(concat(':',sigler(sg2j^[jjb])));
if (1<frac) then begin drawstring(' ; f=1/');drwnum(frac);end;
drawstring(' ; ligne n°');drwnum(i);
drawstring(concat(':',sigler(sgli^[i]),' '));
lir(rpb);erasetect(rec0);
moveto(2,20);
write('faut-il afficher une autre courbe O ou N ');readln(rpc);end;
```

Dans la mesure du possible, on s'est appliqué à écrire, au-dessus ou au-dessous du cadre, tout ce qui peut faire connaître la courbe représentée (cf. exemples: §4): nom du tableau de base; mode de tracé adopté; ligne choisie; valeurs extrêmes de l'abscisse, prise telle quelle dans le tableau ou calculée.

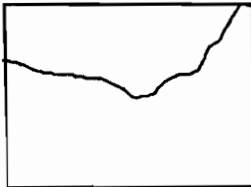
3.4 Fin de l'exécution du programme 'grftab'

```
for j:=0 to cartj do dispose(t1k[j]);dispose(s1i);dispose(s2j);
eraserect(rec0);moveto(2,20);
write('faut-il poursuivre avec un autre tableau O ou N ');
readln(repc);end;end;
readln(rpf);end.
```

Comme de règle, quand l'utilisateur déclare ne plus vouloir d'autre courbe issue du tableau considéré, la mémoire est libérée afin d'accueillir, éventuellement, un autre tableau. (À moins qu'il ne s'agisse de relire le même tableau; mais pour en afficher les lignes autrement).

4 Appendice: deux graphiques, marge et quotient (profil/marge)

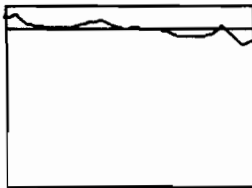
haut= 6.33e-2 ; bas=0 ; ligne de marge de lig n°1: tr3 à lig n°4: tr24



SAPDtx:DatXT ; de la col n°1: N93, à la col n°24: 095; ligne n°5: Marg

À titre d'exemple, deux graphiques, créés pour l'étude à paraître: [OBLIG. LIBAN], sont publiés ici sans aucune retouche, tels qu'ils ont été saisis à l'écran. Au-dessus du cadre de la courbe, est précisé le bloc des 4 lignes cumulées pour calculer la marge; et en dessous, le bloc de colonnes, ou intervalle de temps, représenté: de Novembre 93 à Octobre 95.

haut= 1.15e+0 ; bas=0 ; profil/marge de lig n°1: tr3 à lig n°4: tr24



SAPDtx:DatXT ; de la col n°1: N93, à la col n°24: 095; ligne n°4: tr24

Références bibliographiques

J.-P. & F. BENZÉCRI : "État des modifications apportées aux versions du logiciel MacSAIF", [ÉTAT MacSAIF.]; in *CAD*, Vol.XVIII, n°3, pp. 255-256; (1993);

(Le §1 de cet article donne une notice de la version initiale du programme 'Grftab');

W. SAAB : "Émission des bons du trésor libanais, libellés en dollars et stabilité du marché obligataire", [OBLIG. LIBAN]; à paraître.